

Работа выполнена в рамках задания «Разработать вычислительные средства централизованного мониторинга и прогнозирования паводка» ГПНИ «Научное обеспечение безопасности и защиты от чрезвычайных ситуаций».

#### СПИСОК ЦИТИРОВАННЫХ ИСТОЧНИКОВ

1. Волчек, А.А. Технические средства мониторинга и прогнозирования наводнений / А.А. Волчек, Ю.А. Кузавко, А.Ф. Козак, Д.А. Костюк // Чрезвычайные ситуации: предупреждение и ликвидация: сборник тезисов докладов IV Международной научно-

практической конференции: в 3 т. – Минск, 6–9 июня 2007 г. – Минск, 2007. – Т. 1. – С. 244–247.

2. Козак, А.Ф. Распределенная электронно-информационная система мониторинга и прогнозирования паводка / А.Ф. Козак, Д.А. Костюк, Ю.А. Кузавко, Д.О. Петров // Вестник БрГТУ. – 2008. – №5: Физика, математика, информатика. – С. 104–106.

3. Volchek, A. Electronic system of flood monitoring and visualization / A. Volchek, A. Kozak, D. Kostiuk, D. Petrov // Hydrology: from research to water management. XXVI Nordic hydrological conference. Riga, Latvia, August 9–11, 2010. – Riga: University of Latvia Press, 2010. – P. 66–68.

4. Shuttle radar topography mission. <http://srtm.usgs.gov> 21.07.2011.

Материал поступил в редакцию 25.11.11

#### VOLCHEK A.A., KOSTIUK D.A., PETROV D.O., SHESKO N.N. Flooding area calculation unit for the flood monitoring and prediction system

The algorithm and software implementation for a unit of the floods observation and prediction system is presented, calculating flooding situation based on water level data streamed from control points placed in the river basin. The calculation includes constructing the curved surface of a water mirror and finding the line of its crossing with digital elevation map of the terrain. Proposed method needs no high processing power but correctly models the spread of water over the complex terrain and can be used in distributed calculating networks.

УДК 004.514.62

Костюк Д.А., Костюк К.Л., Дереченник С.С., Тавониус К.А., Шитиков А.В.

### ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ ПЕРЕКЛЮЧЕНИЯ ОКОН В СОВРЕМЕННЫХ ГРАФИЧЕСКИХ ОБОЛОЧКАХ

**Введение.** До недавнего времени в основе графического пользовательского интерфейса всех универсальных операционных систем лежала модель взаимодействия, известная как метафора рабочего стола («Desktop metaphor» или DM). Экран монитора рассматривается в ней как аналогия поверхности рабочего стола, на которой могут размещаться объекты, в первую очередь папки с документами и отдельные документы, которые можно открывать (разворачивать в виде окна), а также вспомогательные приложения [1].

В ходе развития в DM добавлялись усовершенствования, не имеющие объектов-аналогов в реальном мире, в первую очередь средства быстрого запуска приложений и переключения фокуса окон для более полной интеграции с концепцией оконного интерфейса и моделью WIMP («windows-icons-menus-pointer», или окнопиктограммы-меню-указатель). Исторически первым расширением DM для переключения фокуса была док-панель; однако начиная с 90-х годов, в большинстве ОС семейства Windows и Unix ее заменяет панель задач, как более компактное и информативное решение.

Модели WIMP и DM доминировали более 20 лет. Однако в настоящее время из-за возросшей популярности планшетных компьютеров разработчики графических оболочек предпринимают усилия по поиску универсальных решений, пригодных для управления как с помощью мыши, так и средствами сенсорного экрана. В случае успеха такое программное обеспечение должно обладать, помимо универсальности, дополнительным преимуществом: интерфейсы приложений, упростившиеся в ходе адаптации к сенсорному управлению, легче в освоении и в работе, т.к. не перегружают пользователя большим числом деталей.

В версиях графических оболочек, выпущенных в 2011 году, наблюдаются изменения, направленные на отход от DM: альтернативные интерфейсы запуска приложений, отсутствие пиктограмм на рабочем столе, стимулирование пользователей работать с полноэкранными приложениями и, наконец, отказ от панели задач. Послед-

няя либо отсутствует, либо заменяется большей по размеру (и потому более удобной для активации на сенсорном экране) док-панелью, объединяющей ярлыки запуска избранных приложений и кнопки переключения фокуса между приложениями (в отличие от панели задач, ориентированной на переключение фокуса отдельных окон).

В ряде публикаций высказывается предположение о меньшей пригодности измененных графических оболочек для работы в многозадачной среде благодаря заимствованию интерфейсных решений портативных устройств, операционные системы которых частично или полностью построены на принципах невытесняющей многозадачности [2, 3]. Среди рассмотренных изменений на эффективности многозадачной работы оператора должны сильнее всего сказаться перемены, связанные с панелью задач, которая является наиболее активно используемым элементом DM-интерфейса. В настоящей работе предпринята попытка определить особенности влияния данных изменений на работу пользователя в WIMP-среде.

#### Модификации DM в современных графических оболочках.

Для исследования нами были выбраны три графические оболочки последних версий: две не использующие DM, и одна классическая DM-оболочка.

В качестве классической оболочки использовано окружение рабочего стола Plasma Desktop проекта KDE версии 4.7 с панелью задач, расположенной внизу экрана. KDE также содержит элементы, выходящие за рамки DM, однако они опциональны и в стандартном режиме не сказываются на переключении окон.

На роль экспериментальных оболочек выбраны Gnome Shell проекта Gnome 3 и Unity из дистрибутива Ubuntu Linux 11.04. Обе обладают признаками, позволяющими отнести их к пост-DM; однако с т.з. эффективности модели WIMP актуально то, какие средства предложены для переключения перекрывающихся окон.

Костюк Кирим Львовна, к.м.н., врач ГУП «Лечебно-консультативная поликлиника» г. Бреста.

Дереченник Станислав Станиславович, к.т.н., доцент, зав. кафедрой ЭВМ и систем Брестского государственного технического университета.

Тавониус Кирилл Андреевич, ассистент кафедры ЭВМ и систем Брестского государственного технического университета.

Шитиков Алексей Васильевич, студент факультета электронно-информационных систем Брестского государственного технического университета БрГТУ

Беларусь, БрГТУ, 224017, г. Брест, ул. Московская, 267.

Физика, математика, информатика

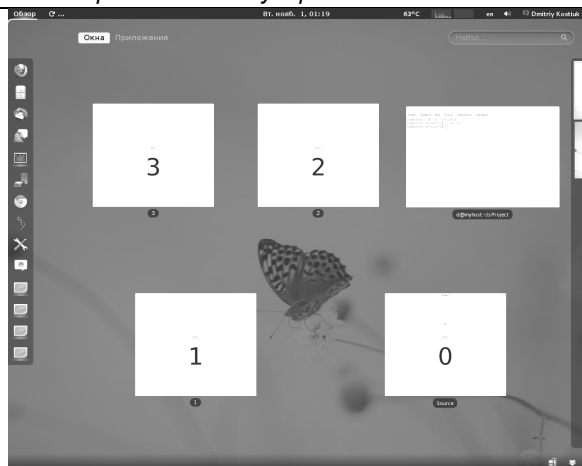


Рис. 1. Миниатюры окон в Gnome Shell

Gnome Shell в стандартном режиме не отображает никакого эквивалента панели задач, предлагая пользователю для манипулирования окнами переключаться в полноэкранный режим обзора (рис. 1) помещением указателя мыши в левый верхний угол экрана либо активацией расположенной там же кнопки «Обзор». Режим обзора содержит дополнительные элементы управления и в т.ч. уменьшенные копии окон для выбора нужного. Поскольку кнопка «Обзор» имеет две бесконечные размерности [1], пользователь WIMP-системы гарантированно достигает ее единственным энергичным движением манипулятора. Но выбор окна требует возврата указателя на достаточно большое расстояние, что уменьшает преимущества угловой активации.

Оболочка Unity предлагает для переключения окон док-панель, расположенную вдоль левой границы экрана, что обеспечивает более короткий путь указателя, сравнимый с расстоянием до панели задач KDE, но при достаточно крупных размерах элементы панели удобны и для активации на сенсорном экране. Панель отображается лишь при наличии свободного места, а при его отсутствии скрывается за левой границей. По классической для док-панели схеме окна одного приложения группируются в одну пиктограмму. В Unity первое нажатие на пиктограмму переключает фокус последнему активному окну группы, а повторное нажатие раскрывает дополнительный экран с миниатюрами окон, временно делая оболочку похожей на режим обзора Gnome Shell, с той разницей, что отображаются окна одного приложения.

Если сенсорное взаимодействие в обеих концепциях сходно и, в силу больших элементов, эффективнее панели задач KDE, то в WIMP-интерфейсе эффект менее однозначен, и для его оценки требуются расчеты и эксперименты.

**Методика тестирования эффективности оконного интерфейса.** Для сравнения выбранных оболочек нами была разработана система, позволяющая протестировать эффективность выполнения оператором однотипных действий в многооконной среде (рис. 2).

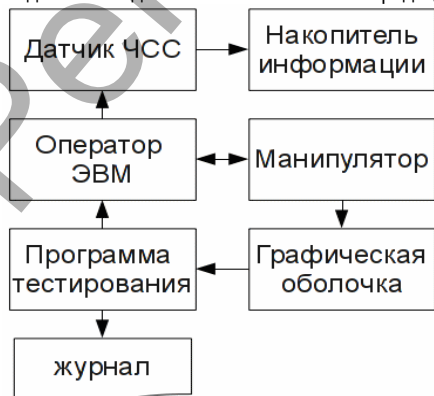


Рис. 2. Структура системы тестирования

Программа тестирования взаимодействует с графической оболочкой, создавая несколько окон, с которыми должен работать оператор. Выделяется фиксированный промежуток времени, за который необходимо выполнить с помощью манипулятора (например, мыши) максимальное количество заданных действий. В качестве основного действия выбрано копирование чисел через буфер обмена.

Система фиксирует в файле журнала сделанные ошибки, а состояние оператора регистрирует датчик частоты сердечных сокращений (ЧСС), что позволяет оценить как скорость и точность выполнения действий, так и физическую нагрузку (мы использовали монитор сердечного ритма Pulse Sonic K920D).

Тестирование проводится в два этапа. На первом этапе пользователь имеет дело с двумя окнами, озаглавленными как «Source» и «Destination». Окна отображаются с размером, обеспечивающим их перекрытие, что принуждает многократно использовать переключение фокуса окон средствами графической оболочки.

Окно «Source» содержит текстовое поле и кнопку «Next», по нажатию которой в текстовом поле генерируется псевдослучайное число. В ходе одной итерации теста оператор должен нажать кнопку «Next», скопировать число в буфер обмена с помощью контекстного меню, затем переключиться в окно «Destination» и вставить число из буфера обмена, также воспользовавшись контекстным меню. По условиям эксперимента клавиатура не используется (кроме специально оговоренных случаев). Окна создаются разными исполняемыми модулями – для оценки эффективности переключения окон разнотипных приложений.

На втором этапе используются  $k$  пронумерованных окон «Destination» и одно окно «Source», которое в дополнение к генерации псевдослучайного числа для копирования также генерирует и отображает крупным шрифтом число  $x \in [1; k]$  – номер окна «Destination», в которое следует произвести вставку (рис. 1). В экспериментах выбрано значение  $k=3$ . Набор действий в итерации теста прежний, но надпись в окне «Source» указывает, в какое из окон «Destination» переключаться на данной итерации. Так выявляется эффективность управления окнами многооконных приложений.

Для повышения требуемой концентрации внимания окна имеют сходную компоновку, включая «фальшивую» кнопку «Next» в окне «Destination».

Перегрузка информацией – одна из причин снижения надежности работы оператора. Защитные механизмы, противодействующие информационной перегрузке, могут приводить, помимо замедления реакции, к пропуску информации, ее ошибочной обработке без последующего необходимого регулирования, пропуску информации в соответствии с некоторой схемой приоритета [4]. Чтобы учесть разнообразие защитных реакций, в файле журнала фиксируются ошибки четырех типов: нажатие на неправильную кнопку, пропуск и дублирование числа, а также неверный выбор окна.

В качестве длительности выполнения одного теста был выбран период 5 минут. Требуемые от оператора действия могут быть отнесены к легкой физической нагрузке [5], что подтверждается характером изменения ЧСС при выполнении теста – ростом на начальном этапе и последующим постепенным снижением. Пятиминутная длительность теста – это время, за которое ЧСС практически выходит на устойчивое состояние, однако на эффективность оператора не успевает повлиять монотонность работы. Таким образом, в каждой из трех графических оболочек оператору отводилось на тестовые задания два раздельных пятиминутных интервала – первый и второй этапы теста.

При оценке результатов рассматривались три типа показателей: быстрота выполнения оператором заданных действий, число и характер допущенных за время прохождения теста ошибок, а также параметры сердечного ритма оператора.

В качестве показателя быстроты выбран *темп прохождения теста*  $\tau$ , равный числу итераций, выполняемых оператором за минуту. С учетом выбранной длительности,  $\tau = N/5$ , где  $N$  – число итераций, которые успел выполнить оператор.

Чтобы абстрагировать данные об ошибках от значения  $N$  для конкретных оператора и оболочки, рассматривалось не общее число ошибок, а среднее число сбоев  $n$ , приходящихся на одну итерацию, и средняя длительность сбоя  $m$ , равная числу затронутых сбоем итераций.

Исходными параметрами при оценке сердечного ритма были сохраняемые монитором среднее, максимальное и минимальное значения ЧСС за время выполнения теста. Очевидно, что переключение окон вносит не такой большой вклад в трудоемкость действий, выполняемых оператором в ходе итерации, как действия над числами. Использование более быстрого способа переключения окон приводит к увеличению числа действий над числами, что в свою очередь повышает утомляемость оператора и снижает число итераций, которые оператор успевает выполнить. Чтобы уменьшить вклад темпа прохождения теста в результат и оценить, насколько различается вклад механизма WIMP при различных условиях проведения теста, мы сравнивали не ЧСС, а интегральный параметр: *напряженность выполнения теста*  $v$ , т.е. среднее за пятиминутный интервал число сердечных сокращений, приходящееся на одну итерацию в конкретном тесте. Это уменьшало вклад, вносимый в показатель разной утомляемостью, характерной для оператора при разном темпе операций.

Оценка напряженности прохождения теста связана с дихотомией «напряженность-утомление», традиционно рассматриваемой при анализе профессиональной деятельности, когда в диапазоне напряженности выделяется оптимальная зона, а также зоны недогрузки и перегрузки, в которых наблюдается снижение эффективности деятельности человека [6]. В ряде случаев слабая помеха положительно воздействует на эффективность оператора (по сравнению с ее отсутствием), и действия по переключению окон, второстепенные к основному заданию, могут рассматриваться в качестве такой помехи. При попадании напряженности в оптимальную зону они должны играть роль микропауз – промежутков между отдельными операциями, которые создают условия для кратковременного отдыха и препятствуют развитию утомления.

Сравнение показателей, полученных для разных графических оболочек, выполнялось отдельно для каждого оператора, с последующим усреднением. Тестовая группа имела равный гендерный состав и включала 22 подопытных в возрасте от 20 до 27 лет.

**Расчет эффективности механизма переключения окон.** Перед анализом экспериментальных результатов была вычислена теоретическая оценка эффективности выполнения тестовых заданий оператором на основе расчета длительности перемещения курсора в одной итерации теста.

Связь времени и точности движения с расстоянием перемещения курсора в модели WIMP рассчитывалась по закону Фиттса, описывающему сенсорно-моторные процессы пользователя [1]:

$$T = a + b \log_2 \left( \frac{D}{W} + 1 \right), \quad (1)$$

где  $T$  – среднее время, затрачиваемое на совершение действия,  $a$  – время начала/остановки движения,  $b$  – параметр типичной скорости устройства,  $D$  – дистанция от точки старта до центра объекта и  $W$  – ширина объекта вдоль оси движения.

В расчетах использовалось разрешение экрана 1280x800 пикселей. Подстановка значений в параметры  $a$  и  $b$  не выполнялась, чтобы не привязывать результат к особенностям конкретного оператора.

Результаты расчета представлены в таблице 1. Для первого этапа тестирования получено минимальное расчетное время для Unity; для KDE было получено чуть большее значение, а время для оболочки Gnome Shell оказалось максимальным. Для второго этапа тестирования благодаря недостаткам док-панели минимальное время получено для KDE. Оболочка Unity переместилась на второе место, а Gnome Shell по-прежнему показал максимальное время

(данные приведены для наихудшего варианта, т.е. при выборе дальнего окна).

Таблица 1. Результаты моделирования

Графическая оболочка	Время итерации	
	1-й этап	2-й этап
KDE	15a + 14.1b	15a + 14.78b
Unity	15a + 13.9b	17a + 14.9b
Gnome Shell	19a + 15.8b	19a + 16.28b

**Результаты тестирования эффективности переключения окон.** Основные результаты проведенных экспериментов представлены на рис. 3.

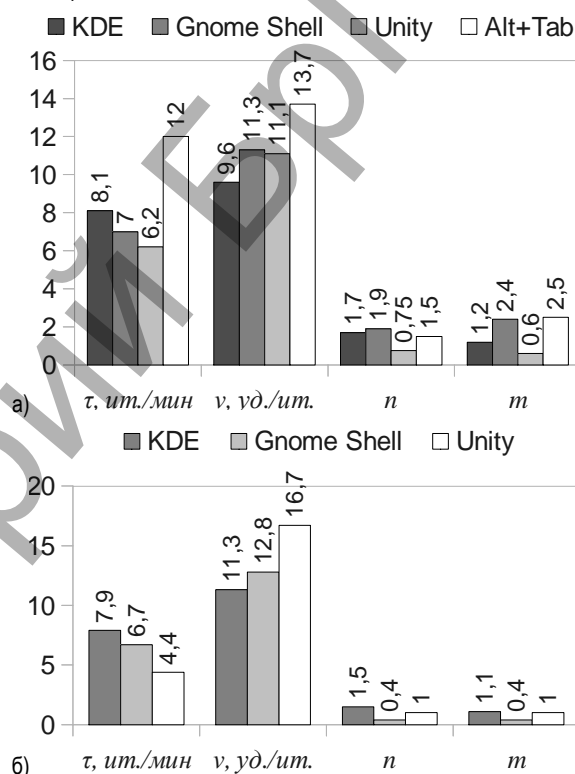


Рис. 3. Результаты тестирования на первом (а) и втором (б) этапах

Часть опытов первого этапа была продублирована с применением клавиатурного переключения окон (нажатием Alt+Tab). Поскольку переключение между двумя окнами выполняется одним и тем же сочетанием клавиш, оператор позиционирует левую руку на соответствующей части клавиатуры, а правая рука располагается на манипуляторе для перемещения курсора. Перемещения манипулятора резко сокращаются, что приводит к увеличению темпа выполнения теста и к некоторому увеличению монотонности действий.

В ходе первого этапа теста получены следующие результаты:

- при использовании клавиатуры были достигнуты максимальные темп и напряженность выполнения теста, а также самые длительные сбои;
- оболочка KDE показала наибольший темп и наименьшую напряженность выполнения теста, а среднее число сбоев в ней оказалось вторым по величине;
- Gnome Shell продемонстрировала средний темп и максимальное число сбоев, а значения длительности сбоя и напряженности выполнения теста оказались вторыми по величине после клавиатуры;

- в Unity были достигнуты минимальный темп выполнения теста, достаточно высокая напряженность (незначительно уступающая Gnome Shell), а также минимальные число и длительность сбоев.

Низкий темп выполнения теста в Unity существенно отличается от расчетных показателей скорости перемещения курсора. Эту разницу можно объяснить более выраженной анизотропией виртуального рабочего пространства WIMP-интерфейса, управляемого мышью. Различная эффективность перемещения мыши в горизонтальном и в вертикальном направлении хорошо согласуется с кинематической моделью костно-мышечного аппарата «кисть-предплечье-плечо». Участие предплечья минимально при горизонтальном перемещении мыши, и эффективность последнего больше зависит от индивидуальных особенностей работы лучезапястного сустава (и, например, должна быть выше у занимающихся настольным теннисом). В Unity, в отличие от других оболочек, горизонтальные перемещения курсора преобладают над вертикальными при переключении окон, а расчетные данные получены для анизотропного рабочего пространства.

Вклад в отставание Unity от других оболочек вносит и меньшая визуальная разница между кнопками выбора приложений. Разработанные тестовые приложения имеют одинаковую пиктограмму. Поэтому в Unity их кнопки в док-панели практически идентичны, в то время как в других оболочках различия хорошо видны: в Gnome Shell одна из миниатюр окон показывает выделенный текст, а в панели задач KDE отображаются заголовки окон.

Высокие, даже при нормировании по темпу прохождения теста, значения ЧСС при использовании клавиатуры подтверждают целесообразность расценивать взаимодействие с механизмом переключения окон графической оболочки при большом числе коротких однотипных операций как помеху, уменьшающую напряженность оператора. Однообразные действия и эмоциональное напряжение (обусловленное подсознательным желанием избежать ошибки) вызывают явление хронического стресса, для которого характерно повышение возбудимости иннервации сердечной мышцы, приводящее к повышению ЧСС. Разнотипные действия, наоборот, уменьшают условия для возникновения хронического стресса.

С точки зрения снижения нагрузки на пользователя, оптимальной оказывается оболочка KDE, т.к. переключение окон в ней наименее трудоемко, а Gnome Shell, требующая наибольшего объема работы для смены фокуса, снижает нагрузку в наименьшей степени. Использование же переключения окон в Unity оказалось слишком трудоемким по сравнению с основными действиями теста.

При выполнении второго этапа (рис. 36) клавиатура не использовалась, а результаты прохождения теста в графических оболочках оказались следующими:

- в KDE были получены наибольший темп, минимальная напряженность и наибольшее число сбоев, как и на первом этапе; однако и длительность сбоев также оказалась максимальной;
- Gnome Shell показала средние темп и напряженность выполнения теста, а число сбоев и их длительность, в отличие от первого этапа, оказались минимальными;
- Unity продемонстрировала минимальный темп (как на первом этапе), максимальную напряженность, среднее число сбоев и высокую (почти как в KDE) длительность сбоя.

По сравнению с первым этапом оболочки KDE и Gnome Shell демонстрируют падение темпа в среднем на 10%, а падение темпа в Unity составило 25% благодаря наибольшей длине цепочки действий по выбору окна.

Значительное число ошибок оператора в KDE при многооконном тесте объясняется меньшим визуальным различием кнопок в панели задач по сравнению с миниатюрами окон (эта особенность тестового приложения хорошо видна на рис. 1). Число сбоев было бы меньше и в Unity, если бы миниатюры исходного окна и окна-приемника были видны одновременно. Кроме того, дополнительный вклад в число ошибок в Unity вносит длинная цепочка действий.

**Заключение.** Результаты показывают, что в целом панель задач остается наиболее эффективным средством переключения окон в WIMP-интерфейсе – по скорости, по нагрузке и по концентрации оператора на выполняемой задаче. При работе с однооконными приложениями наименее быстрым является режим обзора оболочки Gnome Shell, а вертикальная док-панель занимает промежуточное положение. Использование режима обзора также дает наибольшую нагрузку и максимальную плотность ошибок оператора, вопреки тому, что вынесение разработчиками Gnome Shell управления окнами в отдельный режим призвано убрать с экрана все отвлекающие от работы элементы. При управлении многооконными приложениями наименее эффективной оказывается док-панель из-за сокрытия окон приложения под общей пиктограммой. Панель задач демонстрирует максимальную скорость работы, однако при этом возрастает вероятность совершения ошибки оператором. Минимальное же число ошибок достигается в оболочке Gnome Shell (при условии визуальной различимости миниатюр окон).

Таким образом, рассмотренные экспериментальные графические оболочки могут быть рекомендованы преимущественно при работе с единственным однооконным приложением; кроме того, их использование оправдано в ситуации, требующей монотонной работы не менее чем с тремя полноэкранными задачами, когда предъявляются более высокие требования к точности операций, чем к скорости их выполнения.

**Признательность.** Авторы благодарят за участие в экспериментах студентов БрГТУ специальностей «Вычислительные машины системы и сети» и «Промышленная электроника», а также участников международной конференции «LVEE 2011».

#### СПИСОК ЦИТИРОВАННЫХ ИСТОЧНИКОВ

1. Раскин, Дж. Интерфейс: новые направления в проектировании компьютерных систем. – СПб., 2003. – 272 с.
2. Lein, Z.A. Multitasking Differences: WP7, Android, & iOS // Pocketnow. Smartphone reviews, news and video. <http://pocketnow.com/smartphone-news/multi-tasking-differences-wp7-android-and-ios> 20.04.2011
3. Ljubuncic, I. Gnome 3 - this is the end, it seems // Dedoimedo blog. <http://www.dedoimedo.com/> 24.03.2011.
4. Никифоров, Г.С. Самоконтроль как механизм надежности человека-оператора. – Л., 1977. – 192 с.
5. Уилмор, Дж.Х. Физиология спорта и двигательной активности / Дж.Х. Уилмор, Д.Л. Костилл – К., 1997. – 504 с.
6. Забродин, Ю.М. Методологические проблемы функционального состояния человека-оператора // Вопросы кибернетики. Психические состояния и эффективность деятельности. – М., 1983. – С. 3–25.

Материал поступил в редакцию 30.11.11

KOSTIUK D.A., KOSTIUK K.A., DERECHENNIK S.S., TAVONIUS K.A., SHITIKOV A.V. Research of windows switching efficiency in modern graphic shells

The analysis of innovations in modern graphic shells aimed to unify classic cursor control and touchscreen interfaces is carried out. Potential problems are noted for rejecting taskbar in newer versions of GUI. A system to test multi-window interface efficiency is proposed and tested. Based on that calculations and experimental research were carried out for one classical and two experimental graphic shells, including speed, error tolerance and intensity of operator's work at frequent switching of windows. The comparison shows advantages of classical taskbar and limited purposefulness of fullscreen switching of windows miniatures for multi-window applications.